REMARKS

Reconsideration and allowance of the above-reference application are respectfully requested. Claims 13, 20, and 27 are amended, and claims 1-33 are pending in the application.

Claims 13, 20, and 27 are amended to correct an informality: the claims are amended to properly recite the user device as a "user interface device." Hence, the amendment is purely cosmetic, and there is no change to the scope of the claims.

Claims 13, 14, 19, 20, 21, 25, 27, 28 and 32 stand rejected under 35 USC 102(a) as having been anticipated by US Patent No. 6,161,136 to Hyndman et al. This rejection is respectfully traversed for the reasons set forth below.

Each of the independent claims 13, 20 and 27 specify an arrangement in an open protocol network for providing network services to a network-enabled user interface device. In particular, each of the independent claims 13, 20 and 27 specify providing a first network service to the network-enabled user interface device, the first network service generated based on the exchange of service transaction messages between the associated service objects that include a model object, a view object, and a controller object.

Moreover, each of the independent claims 13, 20 and 27 specify that the first network service is supplied based on ***supplying at least one of the service objects*** to the network-enabled user interface device based on the user interaction capabilities of the network-enabled user interface device.

Further, each of the independent claims 13, 20 and 27 specify ***transferring a selected service object*** between any one of a service node, the network-enabled user interface device, and a second network node based on a prescribed condition ***and while maintaining a user-perceived continuous service of the first network service.***

Hence, independent claims 13, 20 and 27 specify the dynamic transfer of service objects, used to perform prescribed operations for a corresponding service, without any interruption in service perceived by the user. Hence, independent claims 13, 20 and 27 not only specify that the network-enabled user interface device can be dynamically configured based on the corresponding user interaction capabilities, but also that a selected service object can be transferred based on a

prescribed condition, ensuring optimum resource allocation for the selected service object, with no interruption in service perceived by the user.

These and other features are neither disclosed nor suggested in the applied prior art.

Hyndman et al. describes partitioning a user interface, which normally resides at a client terminal (see column 1, lines 46-54), into a user interface client (UIC) and a user interface server (UIS) according to a multilevel model-view-controller architecture, that enables deployment of a lightweight user network interface client executable on the customer's host terminal (see, e.g., col. 2, lines 20-31; col. 6, lines 9-12; col. 7, lines 44-45). The user interface server (UIS), however, is deployed on the web server of the service provider (column 6, lines 9-12), and interacts with the user interface client according to the model-view-control architecture of Figure 1B.

As illustrated in Figure 1B, the UIC on the customer's host terminal is illustrated as a top-level object MVC-1 that includes a model cache M-1, view V-1, and mini-controller C-1 (col. 14, lines 16-24): the top-level object MVC-1 also serves as the next-level view object V-2 for the UIS, illustrated as model M-2 (col. 4, lines 7-8), and the controller C-2 which resides on the server (col. 4, line 19). The controller C-2 serves as a data marshaling mechanism provided by the network management system (column 2, lines 31-50; column 4, lines 5-9).

As illustrated in Figure 2 the applied reference simply partitions a user network interface 1 into a UIC 11 (resident at the user's host terminal e.g., web browser and that operates as a view object (V-2)), and a UIS 12 (resident on a web server and that operates as a model object (M-2)). A controller object C-2 also is resident on a server (col. 4, lines 17-20).

Consequently, col. 2, lines 31-50 simply describes that the user interface is partitioned into a client layer executable on the customer's host terminal, and a server layer provided on a separate server as part of a network management system according to a multilevel model-view-controller architecture. Column 4, lines 10-16 simply describes the UIC as a view object V-2 (which interacts with the model object M-2 on the web server) that includes the cache model M-1 (plus the controller C-1, and the view V-1). Column 6, lines 52-57 simply describes the controller object C-1 of the UIC 11 responding to a client action by sending a command event

to the UIS 12, and column 7, lines 1-7 simply describes the controller object C1 initiates generic user feedback to notify the user that the command is being processed.

As apparent from the foregoing, Hyndman et al. is solely concerned with partitioning the user interface into client-server portions using a multilevel model-view-controller architecture, and provides no disclosure or suggestion whatsoever of ***transferring*** a selected service object ***between*** any one of the service node, the network enabled user device, and a second network node based on a prescribed condition, ***and while maintaining a user-perceived continuous service of the first network service***. Rather, the applied reference simply describes partitioning of services on a multilevel model view controller architecture, which presumably would be configured statically before use by the user.

Further, there is no disclosure or suggestion of any supposed installation of a service object into the network enabled user device; regardless, any such installation ***still*** would neither disclose nor suggest the claimed feature of ***transferring*** a the selected service object ***while maintaining the user-perceived continuous service***. Hence, there is no disclosure or suggestion whatsoever of transferring any one of the service objects from one device to another device, as claimed.

Hence, the §102 rejection should be withdrawn because it fails to demonstrate that the applied reference discloses each and every element of the claim. See MPEP 2131. "The identical invention must be shown in as complete detail as is contained in the ... claim." Richardson v. Suzuki Motor Co., 868 F.2d 1226, 1236, 9 USPQ2d 1913, 1920 (Fed. Cir. 1989). "Anticipation cannot be predicated on teachings in the reference which are vague or based on conjecture." Studiengesellschaft Kohle mbH v. Dart Industries, Inc., 549 F. Supp. 716, 216 USPQ 381 (D. Del. 1982), aff'd., 726 F.2d 724, 220 USPQ 841 (Fed. Cir. 1984).

For these and other reasons, the §102 rejection of independent claims 13, 20, and 27 should be withdrawn.

Claims 1-3 and 7-9 stand rejected under 35 USC 103 (a) in view of Hyndman et al. and US Patent No. 5,926,177 to Hatanaka et al. This rejection is respectfully traversed.

As described above with respect to claims 13, 20, and 27, independent claims 1 and 7

specify an arrangement in a network-enabled user interface device, where at least one service object (e.g., a model object, a view object, or a controller object) for a first network service is ***received via the open protocol network***, and the network-enabled user interface device executes the at least one service object for the first network service. Hence, the network-enabled user interface device is able to receive the at least one service object via the open protocol network, enabling the network-enabled user interface device to be dynamically configured to provide a desired network service (the "first network service") for the user.

As admitted in the Official Action, Hyndman et al. does not teach the termination of the received one service object based on reception of a second service object for a corresponding second network service. Further, as described above, Hyndman et al. neither discloses nor suggests that the network-enabled user interface device ***receives*** a service object via the open protocol network; rather, Hyndman et al. is solely concerned with partitioning the user interface into client-server portions using a multilevel model-view-controller architecture. There is no disclosure or suggestion whatsoever of the network-enabled user interface device being capable of receiving multiple service objects (e.g., the at least one service object and the second service object) for first and second network services, respectively, as claimed.

Hatanaka et al. provides no disclosure whatsoever for the claimed selective termination of the received one service object providing the first network service based on reception, via the open protocol network, of a second service object for ***a corresponding second network service***. Rather, Hatanaka et al. simply teaches at column 2, lines 7-10 that old display data is closed as new graphical information is presented. Further, Hatanaka et al. teaches away from the claimed invention by describing at col. 2, lines 11-25 that changing the view object for the same network service is to be avoided: as described on page 3, lines 15-18 of the specification, use of the model view controller architecture typically has been limited to instances where GUI interfaces may be utilized for different hardware configurations.

Hence, Hatanaka et al. teaches a "ViewProxy" that appears to the Model object and Controller object as the same View object, but which manages a "collection of views", where a user can switch between the views using a windowing system to show active windows and hide

inactive windows (see col. 2, lines 50-58, Abstract). Hence, any change in an object state causes the ViewProxy (which is a single view object relative to the model object and controller object) to notify all views, although only the active views of the windowing system will display the changes (see, e.g., col. 4, lines 4-18).

Consequently, the hypothetical combination of Hyndman et al. and Hatanaka et al. neither discloses nor suggests ***receiving*** first and second service objects ***for the respective first and second network services***, where one service object ***for the first network service*** is selectively terminated based on reception, via the network interface, of the second service object for the corresponding second network service. Hence, the hypothetical combination neither discloses nor suggests: (1) receiving the one and second service objects via the open protocol network; or (2) selectively terminating the one service object based on reception of the second service object for a corresponding second network service.

An evaluation of obviousness must be undertaken from the perspective of one of ordinary skill in the art addressing the same problems addressed by the applicant in arriving at the claimed invention. Bausch & Lomb, Inc. v. Barnes-Hind/Hydrocurve, 23 USPQ 416, 420 (Fed. Cir. 1986), cert. denied, 484 US 823 (1987). Thus, the claimed structures and methods cannot be divorced from the problems addressed by the inventor and the benefits resulting from the claimed invention. In re Newell, 13 USPQ2d 1248, 1250 (Fed. Cir. 1989).

At best, the hypothetical combination simply uses the same view object to manage multiple views in respective windows, where only active windows will display changes. Hence, there is no disclosure or suggestion in the hypothetical combination of dynamically providing users with ***different network services*** based on reception of associated service objects via the open protocol network.

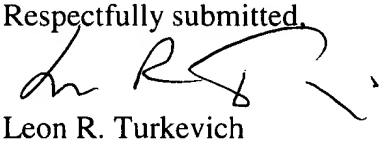
For these and other reasons, the rejection of independent claims 1 and 7 should be withdrawn.

It is believed claims 26 and 33 are allowable in view of the foregoing.

In view of the above, it is believed this application is and condition for allowance, and such a Notice is respectfully solicited.

To the extent necessary, Applicant petitions for an extension of time under 37 C.F.R. 1.136. Please charge any shortage in fees due in connection with the filing of this paper, including any missing or insufficient fees under 37 C.F.R. 1.17(a), to Deposit Account No. 50-1130, under Order No. 95-468, and please credit any excess fees to such deposit account.

Respectfully submitted,



Leon R. Turkevich
Registration No. 34,035

Customer No. 23164
(202) 261-1059
Date: June 1, 2005